

PEMBUATAN SISTEM PENGEMUDIAN OTOMATIS PADA KENDARAAN BERODA DENGAN MODEL PEMBELAJARAN OFFLINE

Indra Nur Harijadi^{#1}, Eru Puspita S.T., M.Kom^{#2}

[#]Jurusan Teknik Elektronika, Politeknik Elektronika Negeri Surabaya
Kampus PENS-ITS Sukolilo, Surabaya

¹indra@student.eepis-its.edu

²eru@eepis-its.edu

Abstrak— Salah satu masalah yang dihadapi pada lingkungan yang tidak terstruktur adalah sulitnya memperkirakan hasil dari setiap aksi secara tepat. Kondisi ini memperkecil kemungkinan menemukan aksi yang akan memecahkan tugas yang diberikan berdasarkan semua kemungkinan yang telah diperhitungkan sebelumnya. Oleh karena itu, proyek akhir ini mencoba untuk membuat sebuah *autonomous mobile robot* yang mampu menghindari halangan dengan algoritma kendali pembelajaran online agar mampu berinteraksi dengan *environment*.

Autonomous mobile robot ini memiliki empat buah sensor ultrasonic digunakan untuk mendeteksi halangan yang ada di sekitar robot. Pusat kontrol dari robot ini ada pada mikrokontroler ATmega32 yang ditanamkan algoritma gerak robot. output dari sistem ini adalah pengaturan motor DC. Hasil dari proyek ini adalah mendapatkan kombinasi gerakan robot untuk menghindari halangan dan mampu beradaptasi dengan lingkungan.

Kata kunci— robot, sensor ultrasonik, SRF04, adaptasi, motor DC, Rute Otomatis.

I. PENDAHULUAN

Perkembangan dunia komputer, terutama dalam mikrokontroler memberikan dampak yang sangat besar terhadap dunia robotika, yang berkembang dengan begitu pesat. Hal ini tentu saja menuntut manusia untuk dapat mengikuti perkembangannya. Banyak pembahasan yang telah dilakukan dalam hal mesin atau alat yang mampu melakukan pembelajaran, dan bagaimana hal ini dapat diaplikasikan pada robot beroda untuk membantu meningkatkan kemampuan operasional. Salah satu hal yang penting dalam perencanaan dan pembangunan pada sistem *mobile* cerdas adalah masalah olah gerak. Olah gerak yang dimaksud terdiri dari perencanaan gerak dan pengambilan keputusan pergerakan untuk menghindari terjadinya tabrakan dalam sebuah lingkungan.

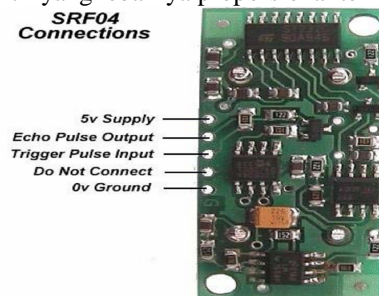
Pada proyek akhir ini yang dimaksud kendaraan beroda adalah sebuah robot beroda. Robot tersebut harus mampu memahami struktur sebuah lingkungan, Cara pengemudian suatu kendaraan otomatis selama ini menggunakan teknik algoritma tertentu yang sudah ditentukan dalam program. Pada judul tugas akhir ini akan dicoba untuk membuat sistem pengendalian yang didasarkan pada proses belajar sendiri. Program akan melakukan proses pembelajaran tertentu

bagaimana untuk bergerak maju, mundur, ke kiri, ke kanan atau kemungkinan lain berdasarkan ada tidaknya halangan. Diharapkan dari proses pembelajaran ini, program dapat menentukan cara bergerak dengan sendirinya jika mengalami atau menemui halangan-halangan. Metode utama yang digunakan di sini adalah jaringan saraf tiruan yang memiliki kemampuan untuk belajar.

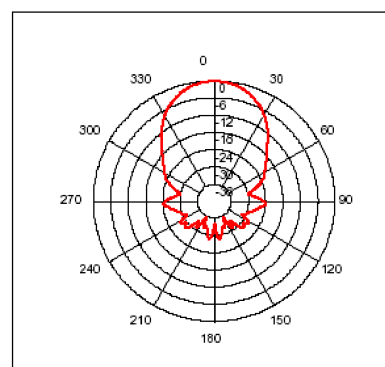
II. TEORI PENUNJANG

A. Sensor Ultrasonik

Sensor ultrasonik yang digunakan adalah Devantech SRF04. Modul ultrasonik Devantech SRF04 mempunyai kemampuan mendeteksi jarak antara 3 cm sampai 3 m dengan frekuensi kerja 40 KHz, mengkonsumsi daya sebesar 30 mA dengan tegangan kerja +5 Volt, *input trigger* berupa pulsa dengan lebar minimal 10 μ s, output pulsa *echo* berupa sinyal TTL positif yang lebarnya proporsional terhadap jarak.



Gambar 1 Koneksi pin SRF04



Gambar 2 Daerah kerja SRF04

B. MIKROKONTROLER

Mikrokontroler adalah suatu piranti yang digunakan untuk mengolah data-data biner (digital) yang merupakan gabungan dari rangkaian-rangkaian elektronik yang dikemas dalam bentuk suatu *chip* (IC). Pada umumnya mikrokontroler terdiri dari bagian-bagian sebagai berikut : Alamat (*address*), Data, Pengendali, Memori (*RAM atau ROM*) dan bagian *input-Output*.

AVR merupakan seri mikrokontroler CMOS 8-bit buatan Atmel, berbasis arsitektur RISC (Reduced Instruction Set Computer). Hampir semua instruksi dieksekusi dalam satu siklus clock. AVR mempunyai 32 register general-purpose, timer/counter fleksibel dengan mode compare, interrupt internal dan eksternal, serial UART, programmable Watchdog Timer dan mode power saving, mempunyai ADC dan PWM internal. AVR juga mempunyai In-System Programmable Flash on-chip yang memungkinkan memori program untuk diprogram ulang dalam system menggunakan hubungan serial SPI.

1) ATmega32

Atmega32 adalah mikrokontroler CMOS 8-bit daya-rendah berbasis arsitektur RISC yang ditingkatkan. Kebanyakan instruksi dikerjakan pada satu siklus clock, Atmega32 mempunyai *throughput* mendekati 1 MIPS per MHz membuat disainer sistem untuk mengoptimasi konsumsi daya versus kecepatan proses. Blok diagram dari mikrokontroler dapat dilihat pada gambar 1.2

Mikrokontroler ATmega32 memiliki bagian-bagian sebagai berikut :

- Saluran I/O sebanyak 32 buah, yaitu Port A, Port B, Port C, dan Port D.
- ADC 10 bit sebanyak 8 saluran.
- Tiga buah *Timer/Counter*.
- CPU yang terdiri dari 32 buah *register*.
- *Watchdog Timer* dengan osilator internal.
- Internal SRAM sebesar 2048 byte.
- Memori *flash* sebesar 32 kBytes.
- Interupsi Eksternal.
- Port antarmuka SPI.
- EEPROM sebesar 1024 byte.
- Komparator analog.
- Port USART untuk komunikasi serial.
- Empat kanal PWM.
- Tegangan operasi sekitar 4,5 V sampai dengan 5,5V.

Beberapa keistimewaan dari AVR AT-Mega 32 antara

lain:

- *Power On Reset* dan deteksi *Brown out* terprogram.
- Sistem mikroprosesor 8 bit berbasis RISC dengan kecepatan maksimal 16 MHz.
- RC Osilator Internal Terkalibrasi.
- Sumber interupsi internal dan eksternal.
- Kapasitas memori *flash* 32 kBytes, internal SRAM sebesar 2048 byte, dan EEPROM (*Electrically*

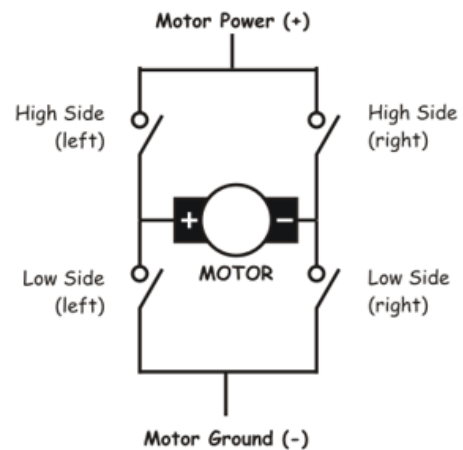
Erasable Programmable Read Only Memory) sebesar 1024 byte.

- ADC (*Analog to Digital Converter*) internal dengan resolusi 10 bit sebanyak 8 saluran.
- Port komunikasi serial (USART) dengan kecepatan maksimal 2,5 Mbps.
- Enam pilihan mode *sleep* menghemat penggunaan daya listrik.

C. MOTOR DC

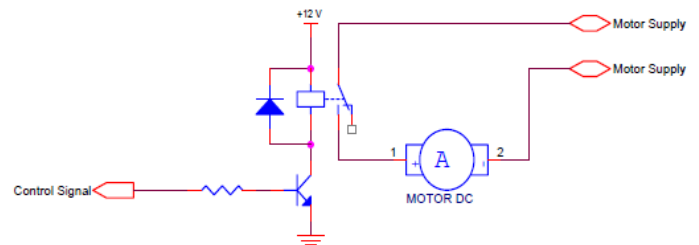
Sesuai dengan namanya, motor DC di-supply dengan tegangan DC (*direct current* = arus searah). Dengan demikian putaran motor DC akan berbalik arah jika polaritas tegangan yang diberikan juga berubah. Motor DC juga terdapat tegangan kerja yang bervariasi, ada yang memiliki tegangan 3V, 6V, 12V dan 24V.

Seperti dijelaskan diatas bahwa pemberian tegangan kerja motor DC diubah, maka arah putaran motor DC juga ikut berubah. Hal ini dapat dijadikan cara untuk mengatur arah putaran motor DC menggunakan saklar. Cara ini merupakan cara pengaturan secara manual, hal ini tidak cocok diterapkan pada sistem robot, karena keluaran dari *port* mikrokontroler tidak cukup memberikan arus yang diperlukan untuk menjalankan motor.



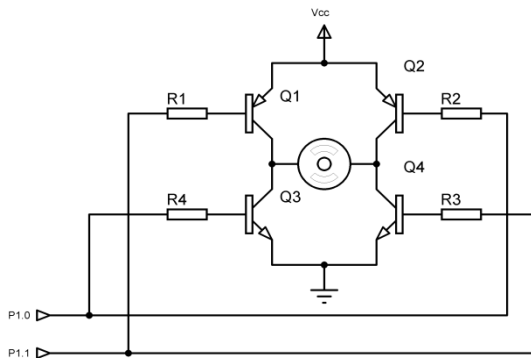
Gambar 3 Pengaturan Arah Motor DC dengan Switch

Cara lain selain menggunakan saklar adalah dengan menggunakan *relay*. Motor DC dihubungkan dengan sumber daya melalui saklar pada *relay* dan *relay* dihubungkan dengan *port* keluaran mikrokontroler melalui *driver* penggerak *relay*.



Gambar 4 Rangkaian Relay untuk Mengontrol Motor DC

Kontrol menggunakan *relay* memiliki kekurangan, karena rangkaian menjadi tidak *solid state* karena *relay* memiliki bagian mekanik. Untuk mengontrol motor DC yang bersifat *solid-state* dapat dipakai rangkaian menggunakan transistor. Transistor disusun sedemikian rupa hingga membentuk huruf H atau yang disebut H-bridge transistor, H-bridge transistor tersusun dari 4 buah transistor dengan memanfaatkan fungsi transistor sebagai saklar, yaitu titik *cut off* dan titik saturasi. Pemilihan transistor yang dipilih dapat mengalirkan arus yang diperlukan oleh motor DC.



Gambar 5 H-bridge Transistor

Dengan metode H-bridge transistor di atas, maka arus yang mengalir ke motor polaritasnya dapat diatur dengan memberikan logika ke transistor Q_1 dan Q_2 . Pengaturannya seperti pada tabel berikut.

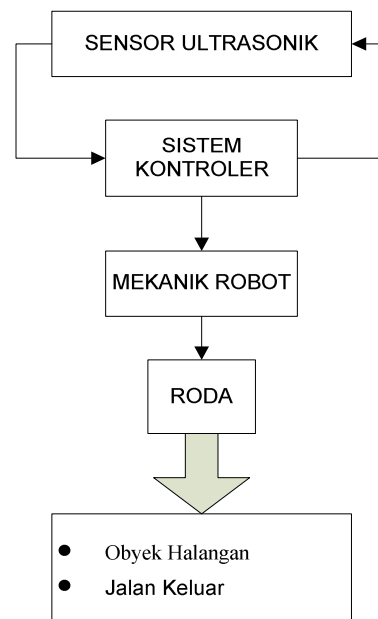
TABEL I
Tabel Kebenaran Keaktifan Motor

P1.0	P1.1	Motor
0	0	Berhenti
0	1	Putar Kanan
1	0	Putar Kiri
1	1	-

Kondisi *High* (1) untuk semua *input* tidak diijinkan, karena akan mengakibatkan semua transistor aktif dan akan merusak transistor karena secara otomatis arus dari kolektor Q_4 dan Q_3 langsung mengalir ke Q_2 dan Q_1 sehingga arus akan sangat besar tanpa melalui beban.

III. PERANCANGAN DAN PEMBUATAN SISTEM

Secara umum konfigurasi sistem dari Robot ini terdiri dari input, kontroler dan output. Dari sisi masukan (*input*) terdiri empat sensor ultrasonik, Kontroler yang digunakan adalah mikrokontroler ATmega32. Dari sisi keluaran (*output*) terdapat driver motor.

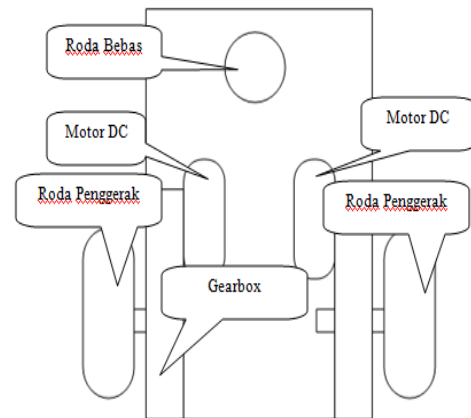


Gambar 6 Blok diagram sistem proyek akhir

D. Perancangan Mekamik

Badan robot dibentuk dari bahan acrylic dengan tebal 3 mm dan disusun tiga tingkat. Pada bagian bawah berbentuk persegi panjang dan terdiri dari bagian penggerak *mobile* robot yang terdiri dari motor DC, gear box dan roda.

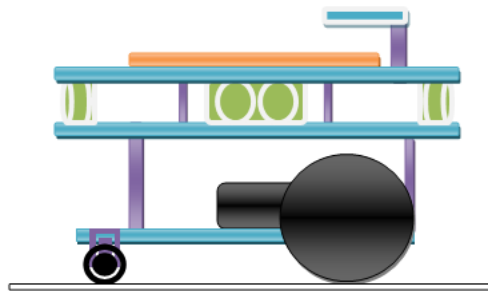
Roda yang digunakan pada robot adalah roda yang biasa digunakan pada mainan dengan diameter roda 3,5cm dan lebar 1cm dengan bahan plastik karena roda jenis ini murah dan mudah dipeoleh di pasaran lengkap dengan *gearbox*.



Gambar 7 Bagian Bawah Robot (tampak atas)

Pada tingkat yang kedua terdiri dari rangkaian mikrokontroler ATmega32, empat buah sensor ultrasonik devantech SRF04. Pada Badan robot tingkat ini juga dibuat dari bahan acrylic 3 mm dengan diameter 15 cm

Pada bagian tingkat ketiga digunakan sebagai tempat LCD yang digunakan untuk memonitor jarak penghalang yang dihasilkan dari sensor ultrasonik



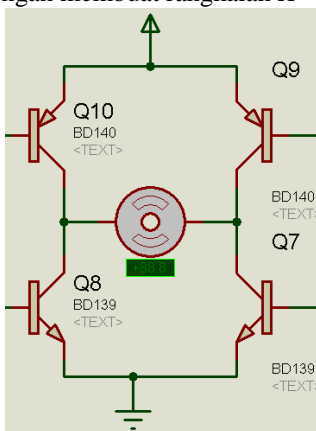
Gambar 8 Ilustrasi robot tampak samping

E. Sensor SRF04

Sistem sensor yang digunakan sebagai input dari proses pengontrolan robot. Pada robot ini dipakai sensor ultrasonik yang berfungsi. Fungsi dari sensor ultrasonik, adalah untuk mendeteksi halangan dan akan dijadikan panduan untuk menentukan berhenti robot dan menghindari. Sensor halangan ini menggunakan Devantech SRF04 yang dipasang dengan membentuk sudut 90° untuk mendeteksi halangan dari bagian depan, belakang, kiri, dan kanan.

F. Perancangan Driver motor DC

Pada sistem ini, *driver* motor DC yang dirancang mempunyai parameter pengontrolan yaitu kecepatan dan arah putaran. Pengontrolan kecepatan putar dilakukan dengan metode PWM (*Pulse Width Modulation*). Dan control arahnya bisa dilakukan dengan membuat rangkaian *H – bridge driver*.



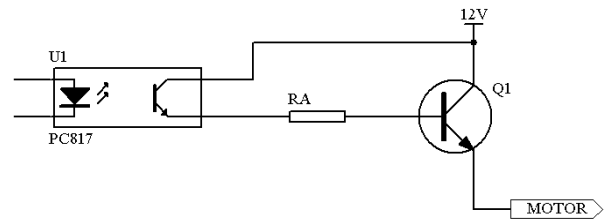
Gambar 9 Rangkaian H-bridge transistor

Dengan metode H-bridge transistor di atas, maka arus yang mengalir ke motor polaritasnya dapat diatur dengan memberikan logika ke transistor Q10 dan Q9. Pengaturannya seperti pada tabel berikut.

TABEL II
Tabel Kebenaran Keaktifan Motor

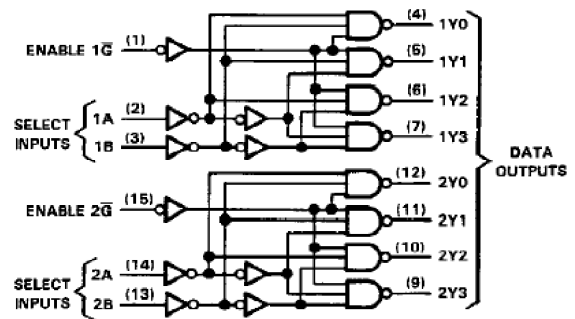
Q10	Q9	Motor
0	0	Berhenti
0	1	Putar CCW
1	0	Putar CW
1	1	-

Pada rangkaian *driver* motor juga digunakan *optocoupler* yang digunakan sebagai pengamanan. Keuntungan menggunakan *optocoupler* adalah agar tegangan yang men-supply mikrokontroler terisolasi terhadap tegangan dan noise yang terjadi pada motor.



Gambar 10 Rangkaian *driver* basis transistor

Rangkaian yang digunakan menggunakan IC 74LS139 sebagai decoder untuk mempermudah program kontrol motor DC



Gambar 11 *logic circuit decoder*

Atmega32 sebagai pengolah sistem. Empat port yang dimiliki oleh chip ini yang digunakan sebagai jalur *bi-directional* dengan pilihan *internal pull-up*. Selain itu ATmega32 juga memiliki fasilitas timer dan interrupt yang cukup baik

Rangkaian I/O dari mikrokontroler mempunyai kontrol direksi yang tiap bitnya dapat dikonfigurasi secara individual, maka dalam pengkonfigurasi I/O yang digunakan ada yang berupa operasi port ada pula yang dikonfigurasi tiap bit I/O. Berikut ini akan diberikan konfigurasi dari I/O mikrokontroler tiap bit yang ada pada masing-masing port yang terdapat pada mikrokontroler.

Port A dari mikrokontroler dikonfigurasi beberapa bitnya yaitu:

- Port A.0 untuk sinyal input dari *sensor ultrasonic1*
- Port A.1 untuk sinyal *trigger* pada *sensor ultrasonic1*
- Port A.2 untuk sinyal input dari *sensor ultrasonic2*
- Port A.3 untuk sinyal *trigger* pada *sensor ultrasonic2*
- Port A.4 untuk sinyal input dari *sensor ultrasonic3*
- Port A.5 untuk sinyal *trigger* pada *sensor ultrasonic3*
- Port A.6 untuk sinyal input dari *sensor ultrasonic4*
- Port A.7 untuk sinyal *trigger* pada *sensor ultrasonic4*

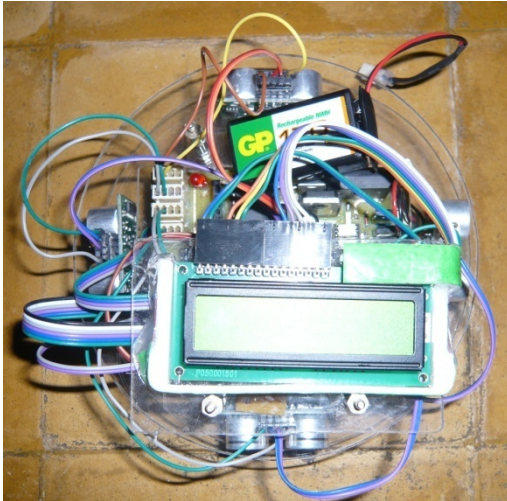
Port C dari mikrokontroler dikonfigurasi beberapa bitnya yaitu:

- Port C.1 untuk rem motor kiri

- Port C.2 untuk *direction* motor kiri
- Port C.3 untuk *enable* motor kiri
- Port C.4 untuk rem motor kiri
- Port C.5 untuk *direction* motor kanan
- Port C.6 untuk *enable* motor kanan

Port D dari mikrokontroler dikonfigurasi sebagai Port untuk LCD yang digunakan untuk memonitoring jarak penghalang.

Dari keseluruhan rancangan dapat dibuat menjadi sebuah rangkaian sistem.



Gambar 12 Keseluruhan sistem(tampak atas)

IV. PENGUJIAN DAN ANALISA

A. Pengujian Sistem Minimum

Pengujian ini dilakukan untuk mengetahui desain bisa bekerja dengan baik mengetahui parameter yang digunakan. Dilakukan dengan program sederhana untuk membuat LED berkedip dengan waktu delay tertentu, dan berikut potongan program yang digunakan.

```
#include <mega32.h>
#include <delay.h>
// Declare your global variables here
#define led PORTA.0

while (1)
{
    led=1;
    delay_ms(500);
    led=0;
    delay_ms(500);
}
};
```

B. Pengujian Sensor SRF04

Pengujian ini dilakukan untuk mengetahui keakuratan sensor dalam mendeteksi jarak halangan. hal ini dapat dilakukan dengan cara menggunakan algoritma untuk menentukan jarak pada persamaan 4.1 atau 4.2.

$Jarak = (Lebar\ Pulsa / 29.034 \mu s) / 2$ (dalam cm)(4.1)
atau

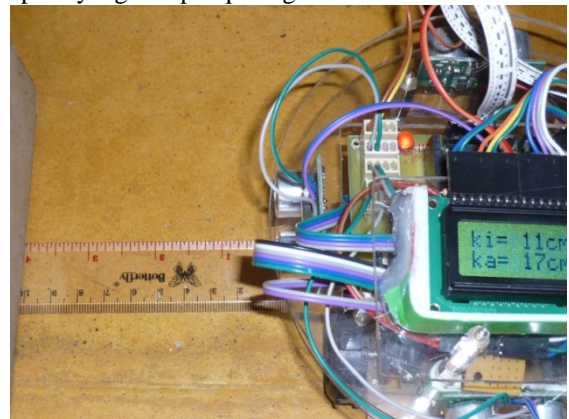
$Jarak = (Lebar\ Pulsa \times 0.034442) / 2$ (dalam cm)(4.2)

Karena $1/29.034 = 0.034442$

Setelah diketahui algoritma untuk menentukan jarak, kemudian algoritma tersebut harus diimplementasikan pada program, dalam hal ini menggunakan *cvavr*.

```
//-----Baca UltraSonic Depan-----
----
unsigned char us_depan(){
    count=0;
    //pemberian pemicu berupa pulsa positif
    minimal 10us
    PORTA.5=1;
    delay_us(15);
    PORTA.5=0;
    //menunggu hingga ECHO = 1
    while(PINA.4==0){};
    //mencacah untuk mengukur waktu lamanya pulsa 1
    dari ECHO
    while(PINA.4==1)
    {
        count++;
    };
    //mengkonversikan hasil cacahan menjadi jarak
    dalam cm
    jrk=(count*0.034442)/2;
    return jrk;
}
```

Setelah program telah berhasil dibuat maka langkah selanjutnya adalah mendownload program tersebut ke dalam mikrokontroler dan dilakukan pengujian untuk mendeteksi jarak seperti yang nampak pada gambar 13.



Gambar 13 Hasil pengukuran jarak SRF04

C. Pengujian Motor

Pada pengujian ini membuat algoritma *PWM* (*Pulse Width Modulation*). Program digunakan untuk mengatur kecepatan motor DC agar gerak robot baik berjalan maju, mundur, belok kanan dan kiri dapat sesuai dengan rancangan sistem.

Pertama yang dilakukan adalah mendefinisikan konfigurasi port agar mudah dalam penamaannya.

```
//KONFIGURASI PIN motor kiri
#define enki PORTC.3
#define dirki PORTC.2
#define remki PORTC.1

//KONFIGURASI PIN motor kanan
#define enka PORTC.6
#define dirka PORTC.5
#define remka PORTC.4

unsigned char pwmka,pwmki,puls;
```

Kemudian penggunaan timer yang digunakan sebagai pengatur besarnya PWM.

```
// Timer 2 overflow interrupt service routine
interrupt [TIM2_OVF] void timer2_ovf_isr(void)
{
// Place your code here
puls++;
    if(puls<=pwmki)
        enki=0;
    else
        enki=1;

    if(puls<=pwmka)
        enka=0;
    else
        enka=1;
}
```

Membuat fungsi gerak robot, seperti maju, mundur, belok kanan, belok kiri, maupun berhenti.

```
void maju()
{
    dirka=0;
    dirki=0;
    remka=1;
    remki=1;
}
```

Pada program utama berisi perintah gerak, dalam pengujian ini menggunakan perintah maju.

```
while (1)
{
    pwmka=255;
    pwmki=255;
    maju();
};
}
```

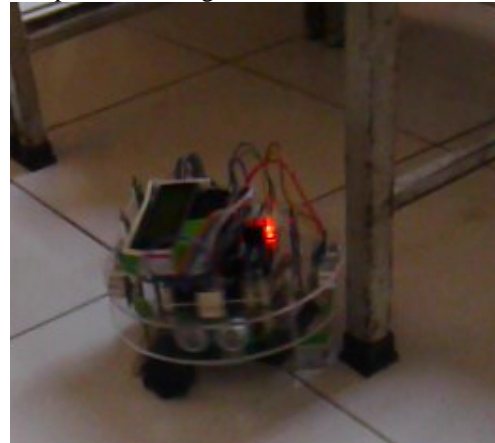
Pada saat nilai PWM yang sama kecepatan putar motor berbeda, hal ini dikarenakan adanya perbedaan kepresisian motor. Penempatan yang motor dan *gearbox* yang kurang presisi akan mengakibatkan robot tidak berjalan lurus meskipun kecepatan motor sama.

D. Pengujian Gerak Robot

Pengujian ini dilakukan untuk mengetahui kemampuan robot dalam menghindari penghalang yang terdapat pada berbagai macam lingkungan.

Pengujian gerak robot ini dilakukan dengan cara menjalankan robot pada berbagai macam lingkungan yang mempunyai berbagai jenis halangan..

- Sebelum diletakan pada sebuah *environment*, robot dibekali dengan kemampuan untuk menghindari penghalang terlebih dahulu.
- Robot diletakan pada suatu lingkungan dengan berbagai macam halangan.
- Penghalang dapat berupa benda dengan permukaan datar, benda dengan tekstur bergelombang maupun benda yang mampu menyerap suara seperti halnya spon.
- Pada saat robot dijalankan pada suatu lingkungan saat menemui halangan yang berupa sudut lancip dalam jarak kurang dari 30cm, maka robot tidak mampu untuk menghindar.



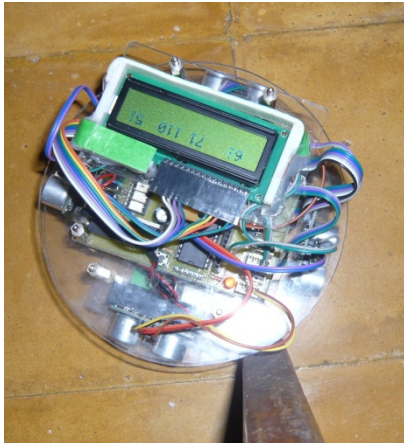
Gambar 14 Robot gagal untuk menghindar pada penghalang bidang lancip.

- Pada saat robot menemui halangan berupa bahan yang dapat menyerap suara (dalam hal ini sofa) dalam jarak kurang dari 30cm, robot tidak mampu menghindar.



Gambar 15 Robot gagal untuk menghindar pada penghalang yang terbuat dari spon.

- Adapun saat dimana sensor gagal mendeteksi halangan dikarenakan halangan berada diantara dua sensor dan tidak memasuki *range* daerah kerja kedua sensor tersebut



Gambar 16 Halangan tidak terdeteksi oleh sensor ultrasonik

Dalam melakukan tugas di berbagai macam lingkungan, robot mengalami berbagai error dengan terjadinya tabrakan karena objek penghalang tidak terdeteksi oleh sensor. Terjadinya error ini dikarenakan kondisi dimana batas daerah kerja sensor tidak terpenuhi. Sehingga tidak didapatkan informasi tentang adanya halangan.

V. KESIMPULAN

Setelah melakukan tahap perancangan dan pembuatan sistem yang kemudian dilanjutkan dengan tahap pengujian dan analisa maka dapat diambil kesimpulan sebagai berikut :

- Sensor ultrasonik SRF04 memiliki tingkat keakuratan 95% saat pengukuran jarak pada bidang datar. Algoritma perhitungan jarak ditentukan oleh medium yang mempengaruhi cepat rambat suara dan kemampuan mikrokontroler dalam mengolah data.
- Kecepatan motor mempengaruhi respon menghindar robot. Bila kecepatan terlalu tinggi maka robot tidak mempunyai kesempatan untuk menghindar.
- Dalam penerapan pada lingkungan terdapat beberapa halangan yang tidak terdeteksi oleh sensor karena halangan yang berada di luar daerah kerja sensor. Sehingga sensor tidak mendapatkan informasi tentang halangan tersebut.

DAFTAR PUSTAKA

- [1] *ATMEL Instruction Set For AVR ATMEGA32*, website : <http://www.atmel.com> diakses pada 29 Mei 2009.
- [2] (2009) <http://shatomedia.com/wp-content/uploads/2008/12/shato-robot-avoidance.pdf>.
- [3] (2009) <http://nugroho.staff.uui.ac.id/2008/12/15/program-mikrokontroler-avr-untuk-sensor-srf04-menggunakan-cvavr/>.
- [4] (2009) <http://gedex.web.id/archives/2008/04/11/pwm-dengan-interrupt-timer/?cp=all#comments>.